

BOUNDARY SCAN ANALYSISFIELD OF THE INVENTION

**[0001]** The present invention generally relates to the generation of boundary scan description language (BSDL)-type files using circuit analysis, and more particularly to the analysis of boundary scan chain implementations.

BACKGROUND

**[0002]** The demand for high-speed, reliable electronic circuits and devices continues to grow at a rapid pace. Concurrently, the complexity of these circuits has also been increasing to meet the needs of a variety of applications. As the complexity and expense of these electronic circuits increases, the testing and analysis of the circuits for a variety of implementations also becomes increasingly important. In addition, accessing certain elements of these circuits has also become more difficult as the elements themselves are scaled smaller and as circuits are formed in increasingly dense arrangements.

**[0003]** Boundary scan analysis is one type of approach used for testing and analyzing circuits, and has been found particularly useful for applications where access to circuit elements for testing is limited. Boundary scan analysis is a methodology allowing controllability and observability of boundary pins of a Joint Test Access Group (JTAG) compatible device via software control. The analysis is typically implemented using the Institute of Electrical and Electronic Engineers (IEEE) standard 1149.1, known as the IEEE Standard Test Access Port and Boundary Scan Architecture. Circuits arranged for boundary-scan analysis typically employ built-in architecture, with pins located at the "boundary" of the circuit and accessible for passing test signals to and from the circuit. Inputs provided via the pins are coupled to various portions of the circuit, depending on the type of

input. Outputs from the circuit are passed to external test circuitry via the pins.

**[0004]** The boundary-scan test architecture provides a means to test interconnects between integrated circuits on a board without necessarily using physical test probes, and can be implemented using a variety of circuits. In most applications, a boundary-scan cell that includes a multiplexer and latches is coupled to each pin on the device. These boundary-scan cells capture data from pin or core logic signals, or force data onto pins. Captured data is serially shifted out via the pins for analysis (e.g., externally compared to expected results). Forced test data is serially shifted into the boundary-scan cells via the data port input pins. The shifting and other boundary-scan operations are controlled, for example, using a serial data path (e.g., a scan path or scan chain).

**[0005]** By allowing direct access to nets, boundary-scan eliminates the need for large numbers of test vectors that are normally needed to properly initialize sequential logic. Tens or hundreds of vectors may do the job that had previously required thousands of vectors. Potential benefits realized from the use of boundary-scan are shorter test times, higher test coverage, increased diagnostic capability and lower capital equipment cost.

**[0006]** BSDL facilitates the description of testability features in components that comply with IEEE Standard 1149, discussed above. This language can be used by tools that make use of those testability features. With a BSDL description of a component and knowledge of IEEE Standard 1149, it is possible to describe data transport characteristics of a particular component that relate to the capture, shifting and updating of data in a particular IC.

**[0007]** While boundary scan analysis has been found very useful, certain aspects thereof have presented challenges. For example, boundary scan analysis relies on programming

language (i.e., boundary scan description language (BSDL)) for operational instructions. However, in some applications, the construction of a correct BSDL file is difficult due to a large number of inputs and outputs and very specific device architecture. Approaches that have previously been used to construct BSDL have typically been unreliable insofar as they depend upon the correct interpretation of relevant device information. For instance, text editor approaches, BSDL editor approaches and synthesis methods that create BSDL files from computer aided engineering (CAE) design data have all exhibited suspect reliability. In addition, automatic analysis methods based upon specific register-transfer level (RTL) description of boundary scan chains or on unmodified or annotated schematics typically fail to address extended circuit features or complex cell designs.

**[0008]** The present invention may address one or more of the above issues.

#### SUMMARY OF THE INVENTION

**[0009]** Various aspects of the invention are directed to approaches for circuit analysis that address the above-discussed challenges related to BSDL generation and others, in some instances bypassing previous error-prone techniques.

**[0010]** In one example embodiment of the present invention, a boundary scan analysis approach involves the determination of various characteristics of a circuit design using the results of test vector simulation. The determined characteristics are then used to generate a comprehensive boundary scan test for the circuit design. With this approach, characteristics of specific circuit designs such as input and output pin connectivity and specific device architecture are readily identified, addressing challenges including those discussed above. In addition, this approach is applicable to both simulated designs and to actual circuits implemented with the circuit design characteristics.

**[0011]** In a more particular example embodiment of the present invention, a general schematic of the circuit design is used to generate an initial set of test vectors. The initial test vectors are applied to boundary scan inputs of the design and a response of the device is detected at boundary scan cells thereof. The response is then used with the general schematic to determine specific characteristics of the circuit design (e.g., extended features or complex designs not available from the general schematic). The specific characteristics are then used to generate more comprehensive test vectors that are applied to the design, with an output response of the device again detected. This process is repeated until sufficient information is collected to describe the Boundary Scan transport characteristics of the circuit design to enable the creation of a testing approach for verifying the design, for instance, by creating a boundary scan description language (BSDL) file.

**[0012]** It will be appreciated that various other embodiments are set forth in the Detailed Description and Claims that follow.

#### BRIEF DESCRIPTION OF THE DRAWINGS

**[0013]** Various aspects and advantages of the invention will become apparent upon review of the following detailed description and upon reference to the drawings in which:

**[0014]** FIG. 1 shows a block diagram of an approach for extracting boundary scan chain information from a design, according to an example embodiment of the present invention;

**[0015]** FIG. 2 is a flow diagram for an approach to boundary scan analysis, according to another example embodiment of the present invention;

**[0016]** FIG. 3 shows an arrangement for boundary scan analysis, according to another example embodiment of the present invention; and

**[0017]** FIG. 4 shows is another flow diagram showing an approach for establishing boundary scan description language

for a device, according to another example embodiment of the present invention.

#### DETAILED DESCRIPTION

**[0018]** The present invention is believed to be applicable to a variety of different circuit analysis approaches and implementations, some of which are described herein. Those skilled in the art will appreciate that various aspects of the invention could also apply to a variety of other circuits and testing approaches.

**[0019]** According to an example embodiment of the present invention, specific characteristics of an electronic circuit used to generate BSDL files are detected using a boundary scan test vector simulation created from a general design schematic of the circuit. Responses (outputs) of the circuit to the test vector simulation are detected and used to identify specific features of the circuit that vary from the general design schematic. For instance, by monitoring output pins of the circuit, a response of the circuit can be mapped to input signals applied to boundary scan chain cells.

Circuit paths, schematic net arrangements and other characteristics of the circuit can thus be identified. As these characteristics are identified, increasingly comprehensive test vectors are generated and used to simulate the design, with corresponding outputs from the simulated design being detected and used to further identify characteristics of the design. Once a sufficiently complete identification of the circuit and a boundary scan chain implementation is obtained, a final test file (e.g., a BSDL file) is created for boundary-scan analysis of the circuit.

**[0020]** FIG. 1 shows a system 100 for generating a test program for a particular circuit design, according to another example embodiment of the present invention. A test program generator 104 parses a netlist 102 to determine architecture characteristics (e.g., type, name and direction of ports) of the design to be tested. Using the architecture

characteristics, the test program generator 104 generates test vectors 106 that are applied to a simulator 108 (i.e., an appropriately programmed computer) that simulates a particular design to which the netlist applies. The test program generator 104 identifies characteristics of the particular design and boundary scan cells thereof as a function of result data 110 generated by the simulator 108 in response to the test vectors 106. For instance, when test vectors selected for generating a reaction from a particular type of circuit element result in an expected response, the circuit element is mapped to an appropriate boundary scan cell or a pin.

**[0021]** The identified characteristics are used with the original netlist 102 to generate new test vectors 106, which are applied to the simulator 108 to generate result data 110 with the new test vectors. The process of applying test vectors and using result data to generate new test vectors is repeated in this manner until selected completion criteria are met for establishing a test program 102. The selected completion criteria may be met, for example, when sufficient information about a boundary scan implementation involving boundary scan cells for the circuit is identified such that a unique test program can be written.

**[0022]** In one implementation, a physical circuit is used in place of the simulator 108 discussed above in connection with FIG. 1. The physical circuit employs circuitry characterized by the netlist 102. The test program generator 104 is connected to boundary scan access ports and pins of the physical circuit to respectively apply the test vectors 106 and detect the result data 110. After sufficient information about the boundary scan implementation of the physical circuit is extracted, the test program generator 104 uses the extracted information and the netlist to create the test program 102 for analyzing the physical circuit.

**[0023]** FIG. 2 shows a flow diagram for generating a test program for a circuit design, according to another example

embodiment of the present invention. The approaches shown and discussed in connection with FIG. 2 below are applicable, for example, with the implementations shown and discussed in connection with FIG. 1 above. At block 202, a design netlist is input and used to generate an initial current set of test vectors. The operation of the design is simulated at block 204 using the current set of test vectors to generate an output, and the output is used to determine information (e.g., circuit characteristics) about the design. If sufficient information about the design has been determined at block 206, the test program is generated from the current set of test vectors at block 210.

**[0024]** If sufficient information about the design has not been determined at block 206, the simulated output is analyzed and used to generate a new current set of test vectors. The new current set of test vectors takes into consideration the determined information and is selected for generating an output from the design to identify further information. The process resumes at block 204 and is repeated until sufficient information for generating the test program is determined.

**[0025]** FIG. 3 shows a test controller/generator 360 used in identifying characteristics of a boundary-scan compatible circuit 300 and establishing a boundary-scan description language file, according to another example embodiment of the present invention. The test controller/generator 360 may, for example, be implemented using one or more of the approaches discussed herein, for example as described below in connection with FIG. 4. In addition, the test controller/generator 360 and the functions it performs may, for example, involve two or more separate units.

**[0026]** The circuit 300 includes internal logic 310 with boundary-scan cells 311-318 coupled thereto. A boundary scan portion of the circuit 300 includes a bypass circuit 320, instruction register 330, data register 340 and TAP controller 350. Test data input (TDI) node 302 receives test

inputs and test data output (TDO) node 304 outputs test data. TDI node 302 and TDO node 304 may, for instance, include a plurality of input or output pins, respectively. Test clock (TCK) node 306 and test mode select (TMS) node 308 respectively receive clock and mode signals from the test controller/generator 360. The bypass circuit 320 selectively bypasses input signals received at TDI node 302 directly to the TDO node 304 and, for example, to other circuits similar to and coupled in series with the circuit 300. A bypass instruction from the test controller/generator 360 allows the circuit 300 to remain in a functional mode while passing data to another circuit to be tested. Optionally, one or more of the circuit components (e.g., TAP controller 350 or instruction register 330) are part of the test controller/generator 360.

**[0027]** The operation of the boundary-scan cells 311-318 is controlled with the TAP controller 350 and the instruction register 330, with the optional use of one or more data registers 340 for various test functions. Under normal operating conditions, the boundary scan cells 311-318 are inactive and allow data to pass to the internal logic 310. Under test conditions, the boundary scan cells 311-318 are active and used to capture response data from the internal logic 310 to test input vectors.

**[0028]** The test controller/generator 360 uses a netlist to generate an initial set of test vectors and applies them to the circuit 300. The test controller/generator 360 uses responses from the circuit 300 to identify characteristics of the circuit that are then used for establishing more comprehensive test vectors (i.e., test vectors utilizing aspects of the identified characteristics). The comprehensive test vectors are then applied to the circuit 300, with responses from the circuit again being used to identify additional characteristics of the circuit. This process of applying test vectors and identifying characteristics of the circuit is repeated until sufficient



information about the circuit 300 is identified for establishing a boundary scan program.

**[0029]** FIG. 4 shows an approach for generating a BSDL file using a general design netlist and test vector simulation to identify specific characteristics of a circuit design, according to another example embodiment of the present invention. The approach shown in FIG. 4 may be applicable, for example, to the circuit 300 shown in FIG. 3. In this regard, reference to selected elements of the circuit 300 is made below.

**[0030]** At block 410, a general device netlist is analyzed to determine general circuit characteristics, such as the number of input/output (I/O) pins, top level pins and pads, and the logic level of particular states. At block 420, a generic test bench is built using the device netlist. The generic test bench involves identifying characteristics such as register existence and/or length (e.g., for instruction, boundary scan, bypass or data registers) and selecting a correct register for particular instructions. For instance, referring to FIG. 3, various characteristics related to the bypass circuit 320, instruction register 330 and data register 340 can be identified at block 420. A device-specific test bench specific to the circuit design for which characteristics are being identified is then built at block 430 using the generic test bench. The device-specific test bench is then applied to the circuit design at block 440, with results being analyzed to determine information about the circuit design.

**[0031]** Using the circuit arrangement shown in FIG. 3 as an example, the following describes an approach for generating information to be analyzed at block 440 with the device-specific test bench simulation. Sample/preload and EXTEST instruction operations are carried out (with the instructions involving testing for the IEEE 1149.1 standard, discussed above). Typical sample/preload instructions set the device being analyzed in a functional mode and connect a boundary

scan register (e.g., data register 340) between input (e.g., TDI 302) and output (e.g., TDO 304) nodes. During sample/preload instruction, the boundary scan register can be accessed to take a sample of functional data entering and leaving the device being analyzed, and to preload test data into the boundary scan register. Typical EXTEST instructions involve setting an external boundary test mode, connecting boundary scan register(s) between input (e.g., TDI 302) and output (e.g., TDO 304) nodes and loading boundary scan cells (e.g., cells 311-318) with test patterns. In addition, input boundary cells are mapped to I/O pins and to particular input instructions, and control and output cells are mapped to I/Os. The correct operation of scan cells and I/O pin connections is also verified (or identified as being incorrect).

**[0032]** The above-discussed and other results are analyzed at block 440 and characteristics of the circuit design are identified therefrom. If information about the circuit design sufficient for generating a BSDL file is available at block 450, a new device-specific test bench is built at block 452. The process of analysis and identification of circuit characteristics then resumes at block 440 with the new device-specific test bench, with new test vectors being applied to the circuit design. If sufficient information for generating the BSDL file is available at block 450, a pin-to-package pinout file for the circuit design is parsed at block 460 and a BSDL file is generated using the analyzed results at block 470.

**[0033]** The BSDL file may then be verified for correctness at block 472. The correctness of the BSDL file may be determined by way of syntax verification and test vector generation, which may be performed by a test service vendor, for example. JTAG test vectors may be used in device simulation or directly applied to a device on a tester (not shown). Test vectors may be used to detect timing or synthesis errors.

**[0034]** Those skilled in the art will appreciate that various alternative computing arrangements would be suitable for hosting the processes of the different embodiments of the present invention. In addition, the processes may be provided via a variety of computer-readable media or delivery channels such as magnetic or optical disks or tapes, electronic storage devices, or as application services over a network.

**[0035]** The present invention is believed to be applicable to a variety of circuit approaches, and has been found to be particularly applicable and beneficial in circuit analysis involving boundary-scan type signals. Other aspects and embodiments of the present invention will be apparent to those skilled in the art from consideration of the specification and practice of the invention disclosed herein. It is intended that the specification and illustrated embodiments be considered as examples only, with a true scope and spirit of the invention being indicated by the following claims.